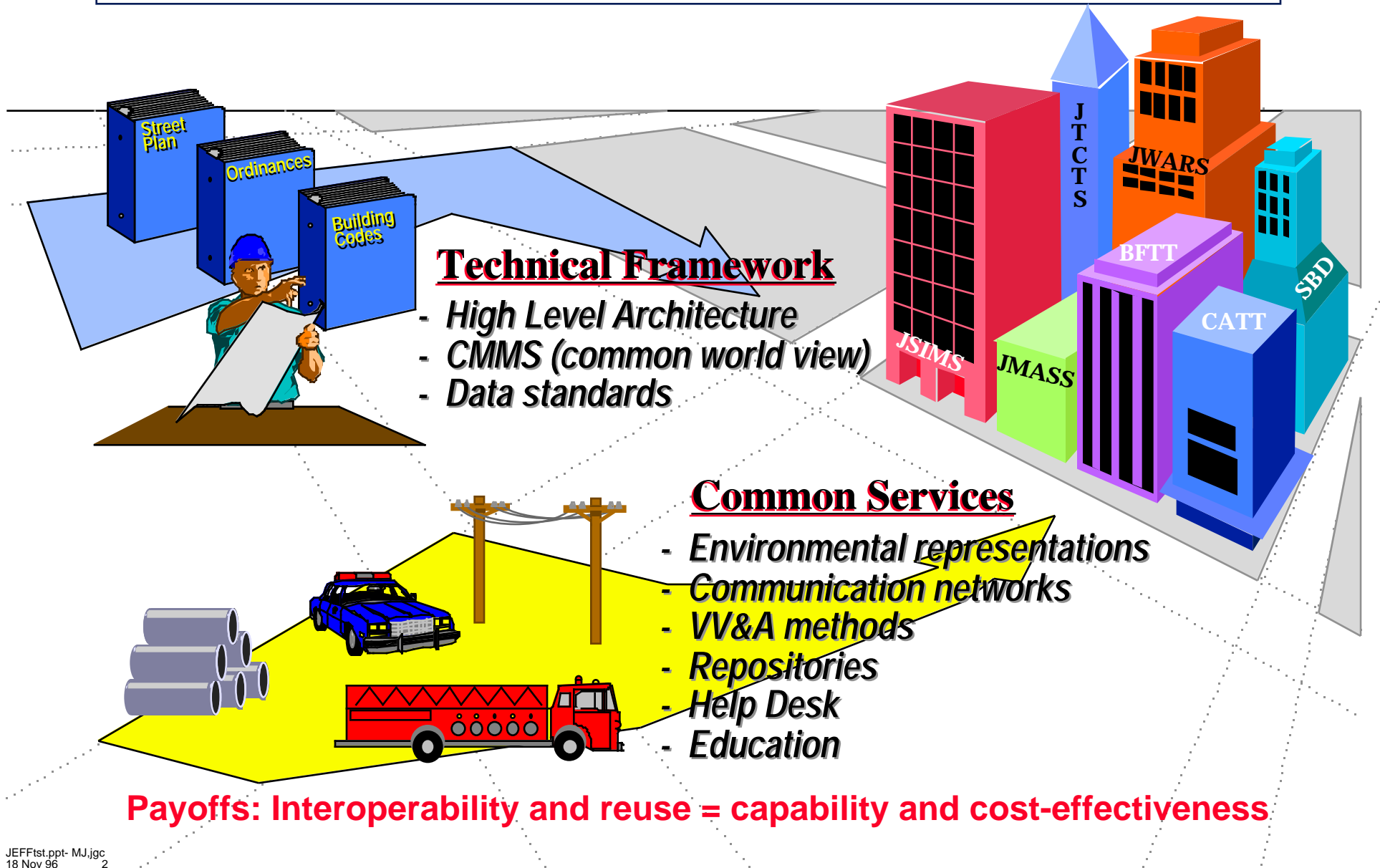




# **Defense Modeling & Simulation Office Conceptual Models of the Mission Space (CMMS)**

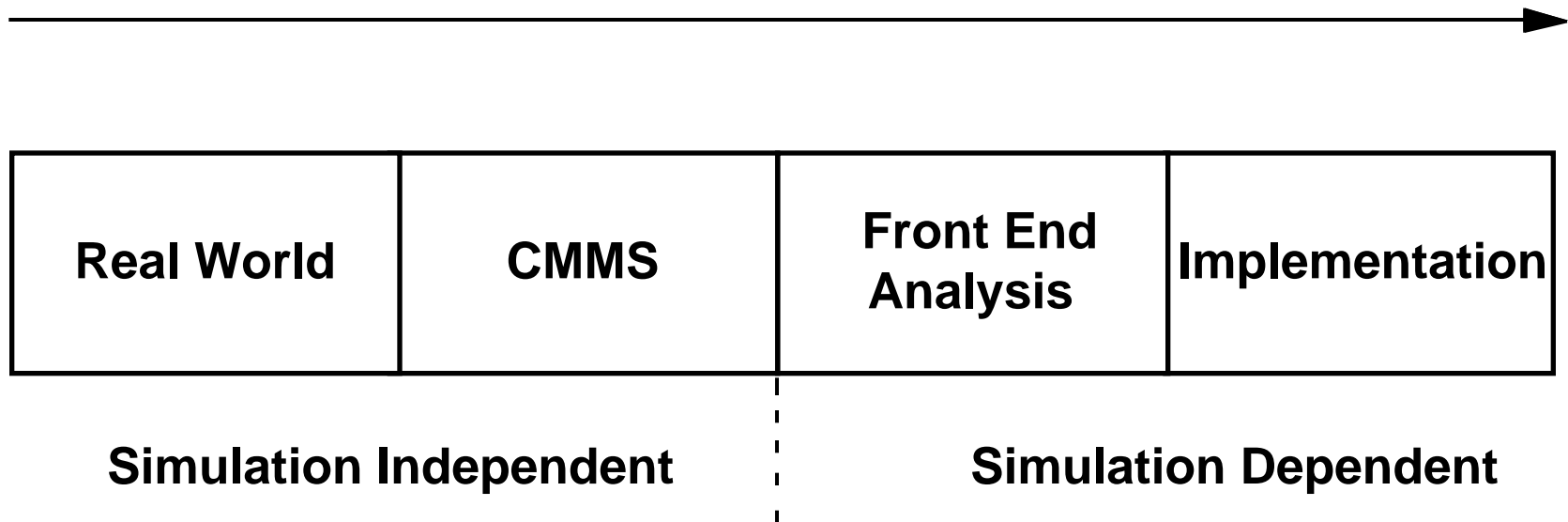
**Col Mark Jefferson  
Deputy Director  
Defense Modeling and Simulation Office (DMSO)  
(703) 998-0660 fax (703) 998-0667  
mjeffers@dmsomil**

# DoD M&S Strategy: An Analogy to City Planning



# Conceptual Models of the Mission Space

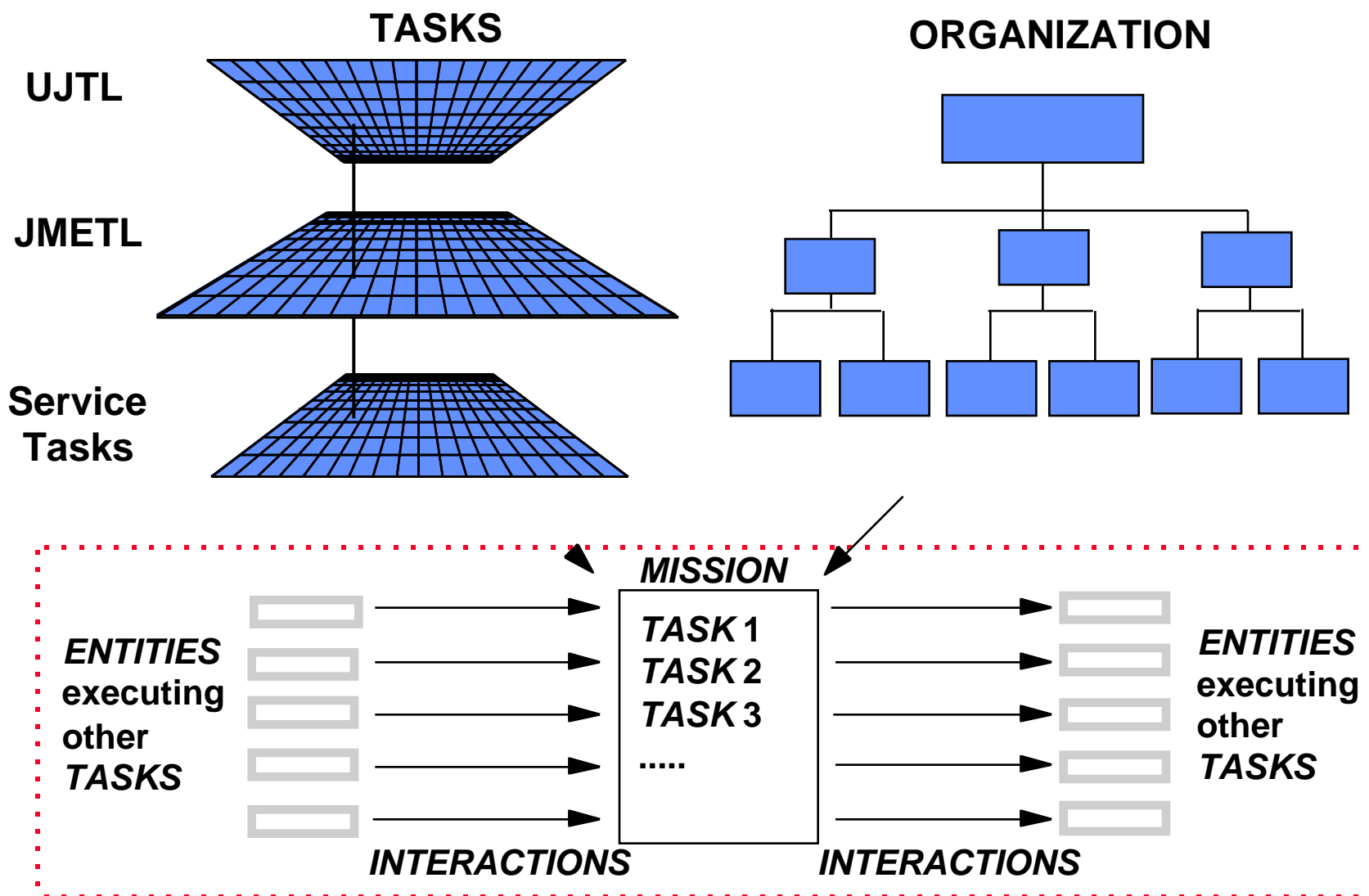
## Simulation Development Process



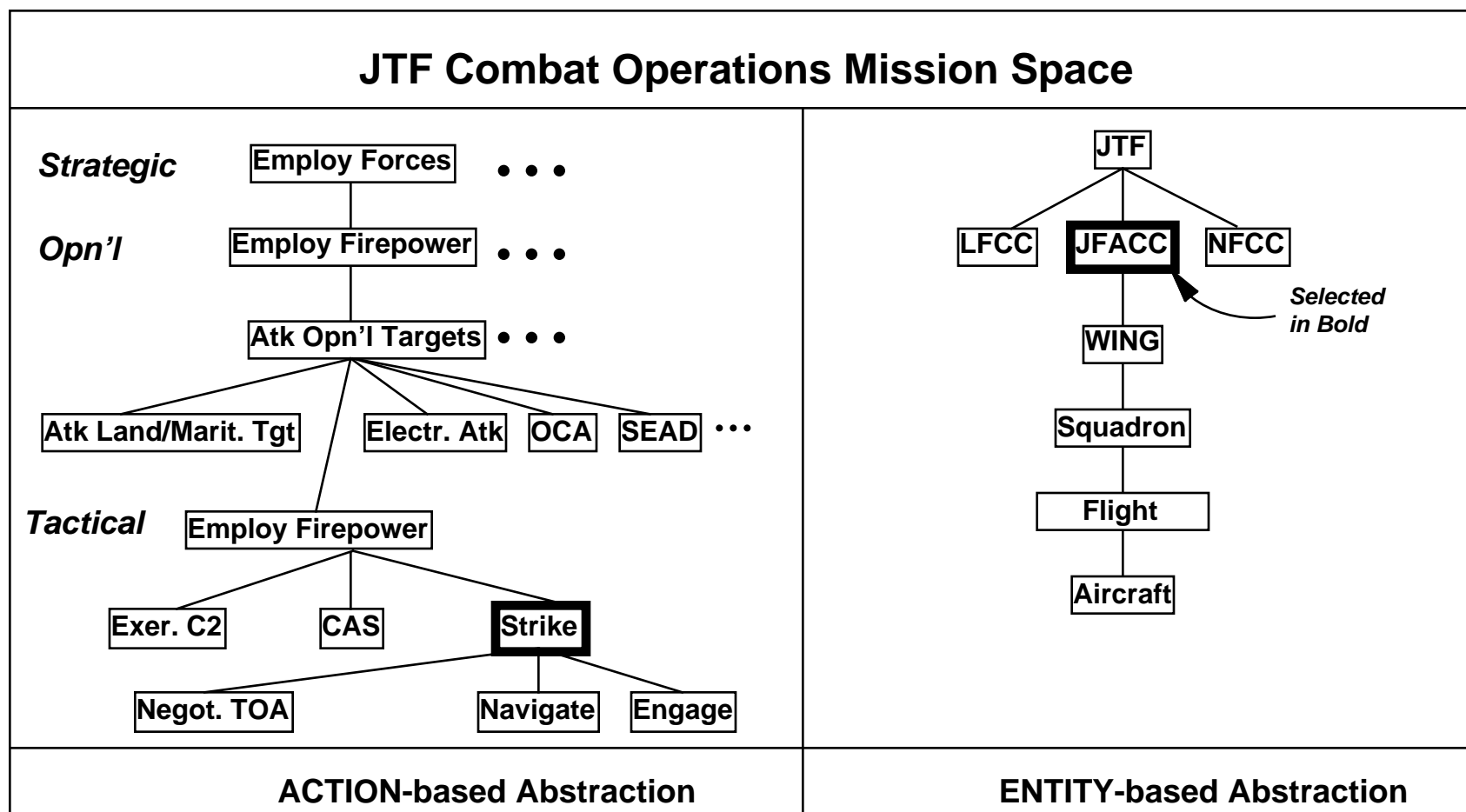
# What is a CMMS?

- **First abstraction of real world**
- **Simulation independent hierarchical description of actions and interactions among various entities associated with particular mission area**
- **Common framework for knowledge acquisition**
  - **Validated, relevant actions and interactions organized by specific task and entity/organization (i.e., ADS's)**
  - **Standard format for expression (i.e., CSS's, DIF's)**
- **Purpose of CMMS is to cost-effectively provide simulation developers (and others) common understanding of real world**
  - **Disciplined procedure by which simulation developer is systematically informed about real world to be synthesized**
  - **Information structure that simulation subject matter expert employs to communicate with and obtain feedback from military operations subject matter expert**
  - **Real world operations basis for subsequent, simulation-specific analysis, design, and implementation decisions**
  - **Singular means for identifying re-use opportunities in eventual simulation implementation by establishing commonality in real world activities**

# Organizational Concept



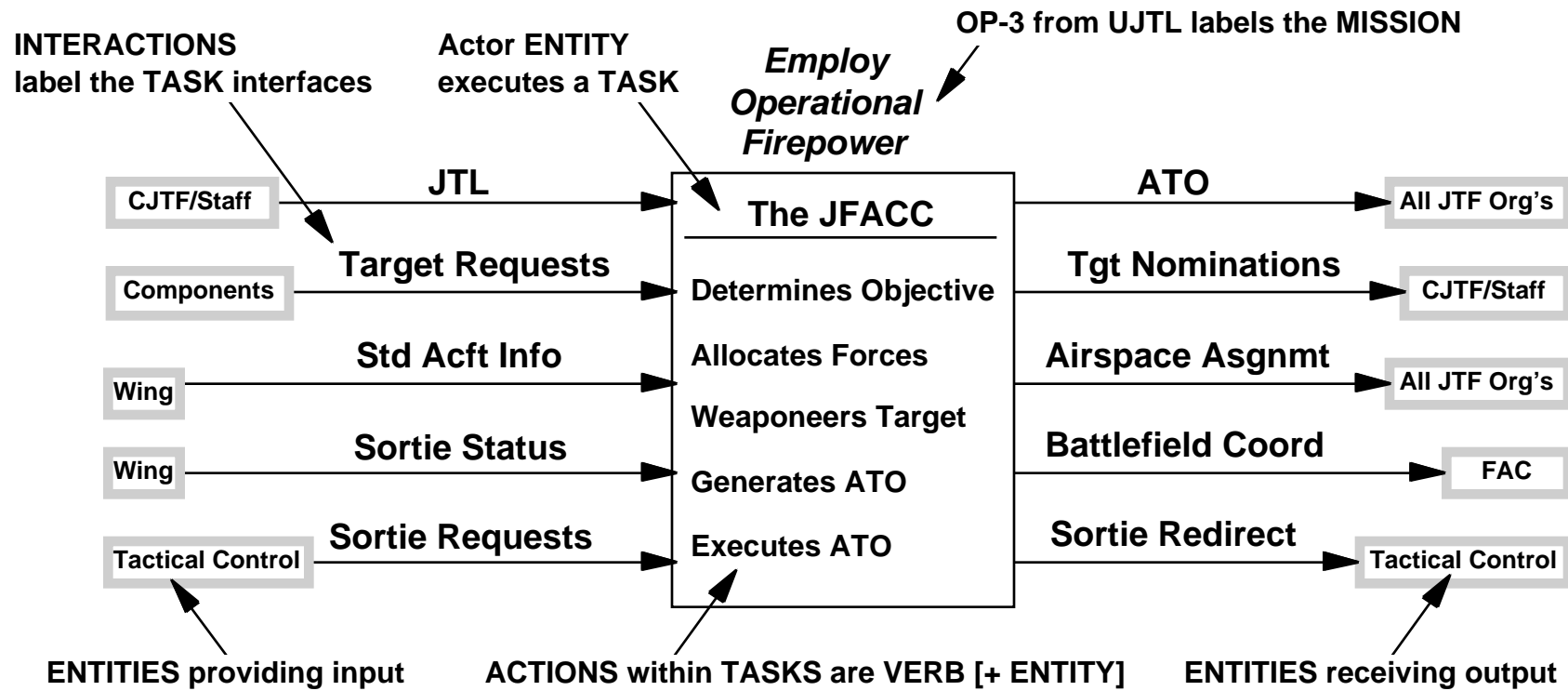
# ACTION and ENTITY Selection Example



***User Selects the ACTIONS Involved and allocates ENTITIES to execute these ACTIONS in the Strike Task by the JFACC ...***

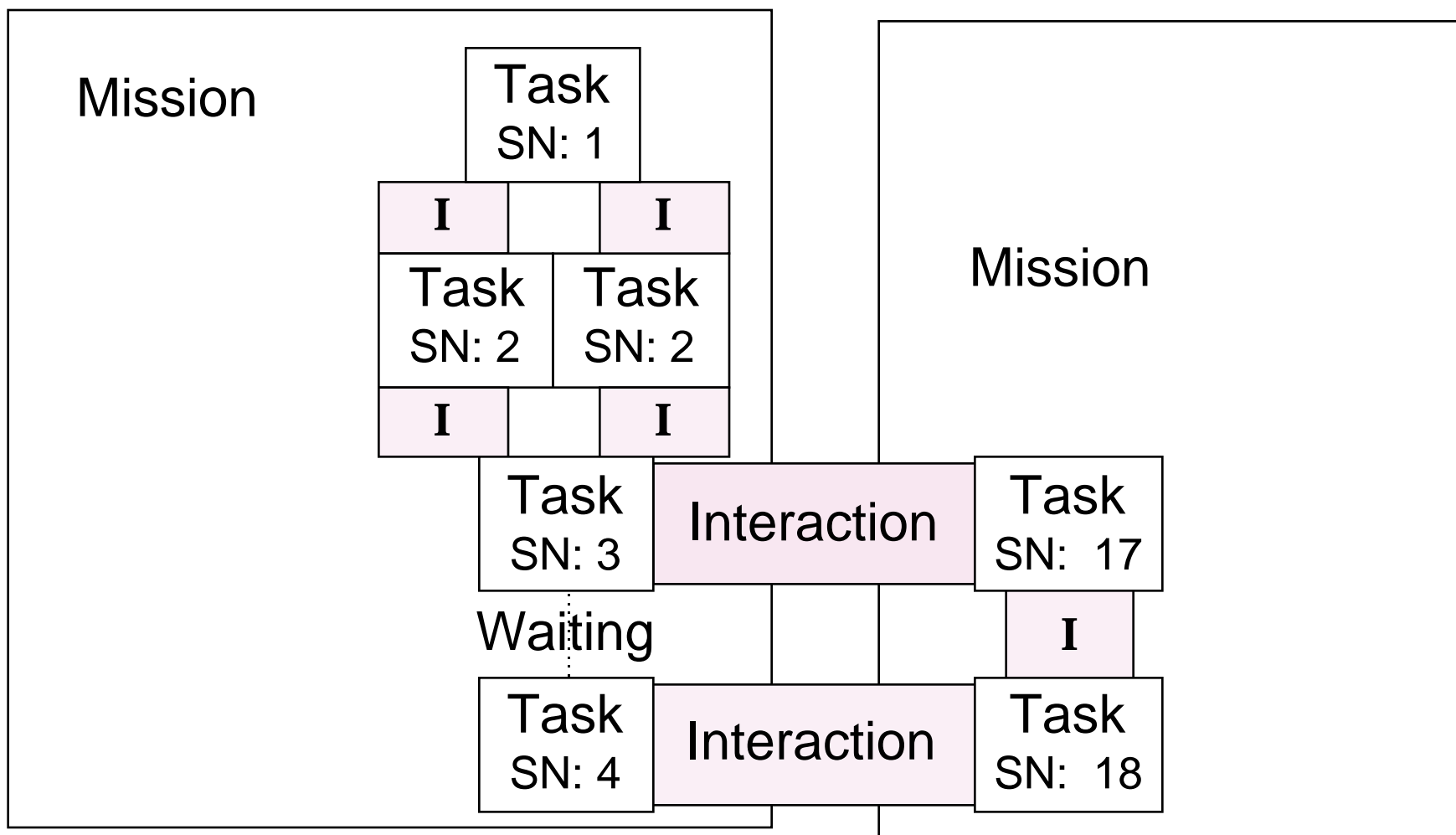
# ENTITIES, ACTIONS, TASKS, and INTERACTIONS

## JTF Combat Operations Mission Space



***Template shows the JFACC tasks  
(e.g., JFACC generates ATO)  
along with inputs, outputs, and their interfaces***

# Mission Threads





# CMMS Technical Framework

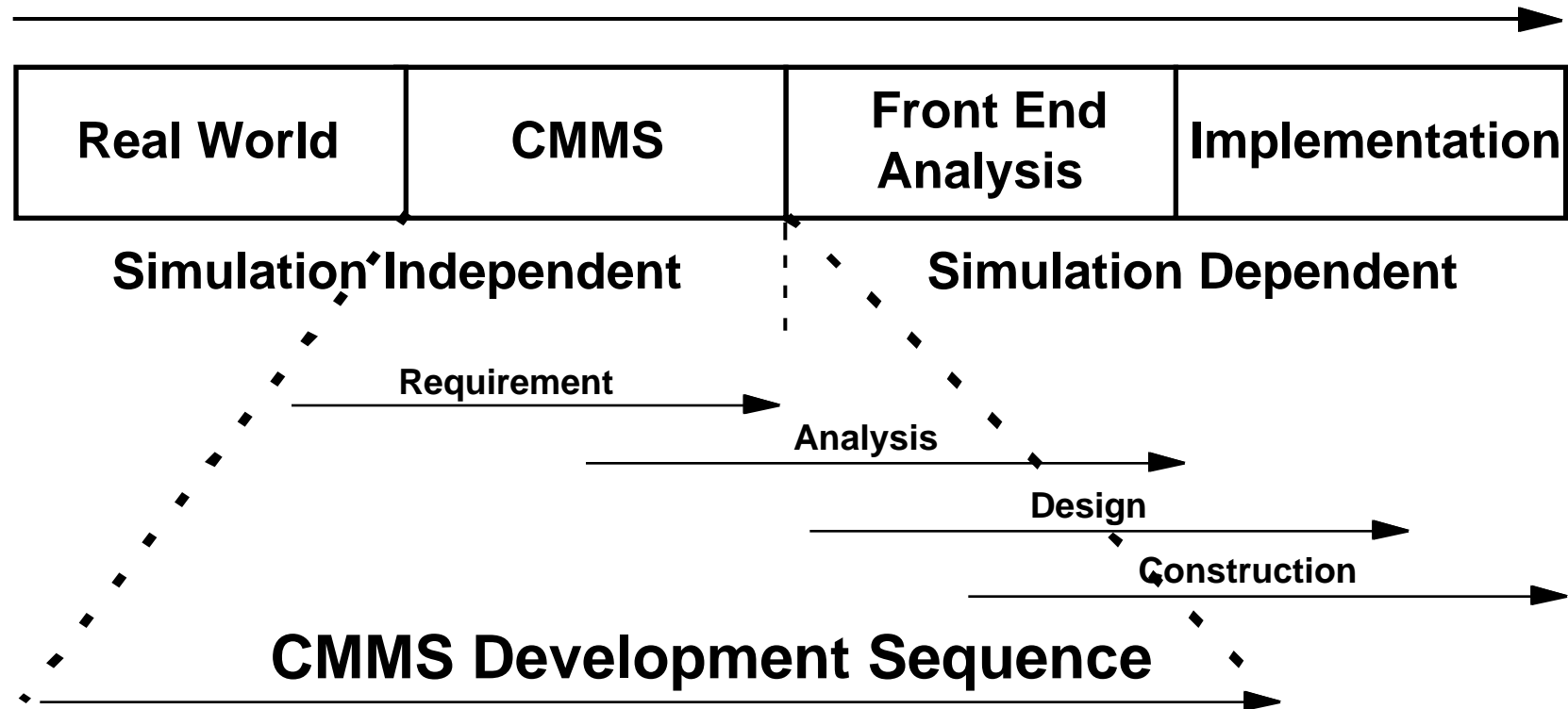
- **The DMSO-sponsored CMMS Technical Framework provides the**
  - technical standards
  - operational procedures
  - information systems infrastructure**required for conceptual model integration within simulation projects and interoperability across DoD programs**
- **The CMMS Prototype Cycle 1 (completed 10/24/96) implementation of CMMS-TF version 0.1 (4/26/96)**
  - 9 standard functions required to create and manage models
  - preliminary Common Syntax and Semantics (CSS)
  - a common Data Interchange Format (DIF)**established the feasibility and the common technical approach**
- **CMMS-TF version 0.2 (draft 11/15/96) is the formal vehicle for implementing the common technical approach**
  - between JWARS and JSIMS
  - between the JSIMS JPO, EA's and DA's, and partner programs**and eventually among all DoD M&S activities**

# CMMS Technical Framework Version 0.2

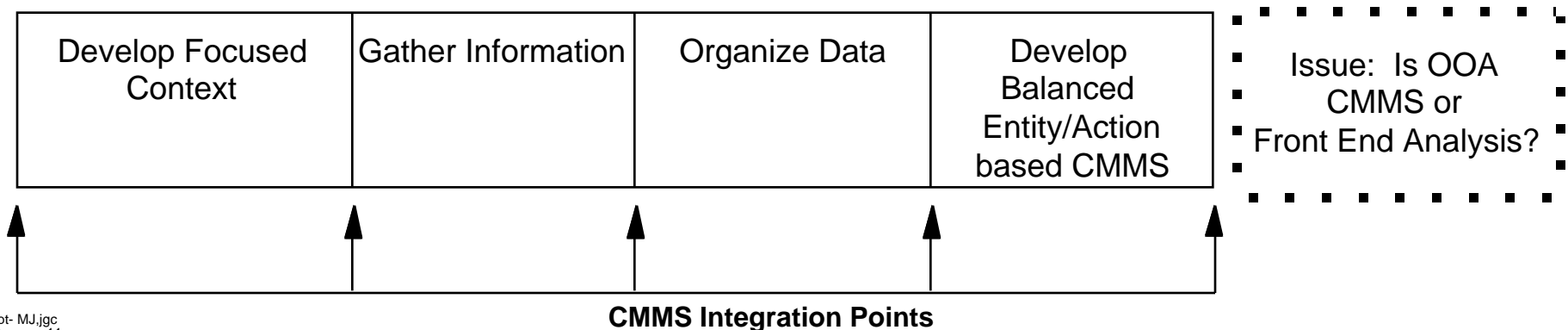
- **Establishes a common CMMS development sequence**
  - develop Mission Threads to provide focused context and identify priorities
  - gather information by registering ADS on MSRR
  - organize data using standard Op-Spec and Use-Case templates
  - construct balance Entity/Action-based CMMS using CSS
- **Note: OO analysis is the end of CMMS and the beginning of Front End Analysis. Some projects may include OO models in CMMS.**
- **Establishes Data-Centric CMMS Integration Standards**
  - extend 9 CMMS functions to 20 Data Standards functions
  - 3-layer CSS (general IS templates, military EATI, CASE-tool style guides)
  - CASE-tool independent DIF
  - formal Interface Design Descriptions (IDD's)

**for scalability within projects and interoperability between programs**
- **Employ the MSRR/CMSRR for**
  - physical access
  - network connectivity
  - data releasability across DoD M&S programs

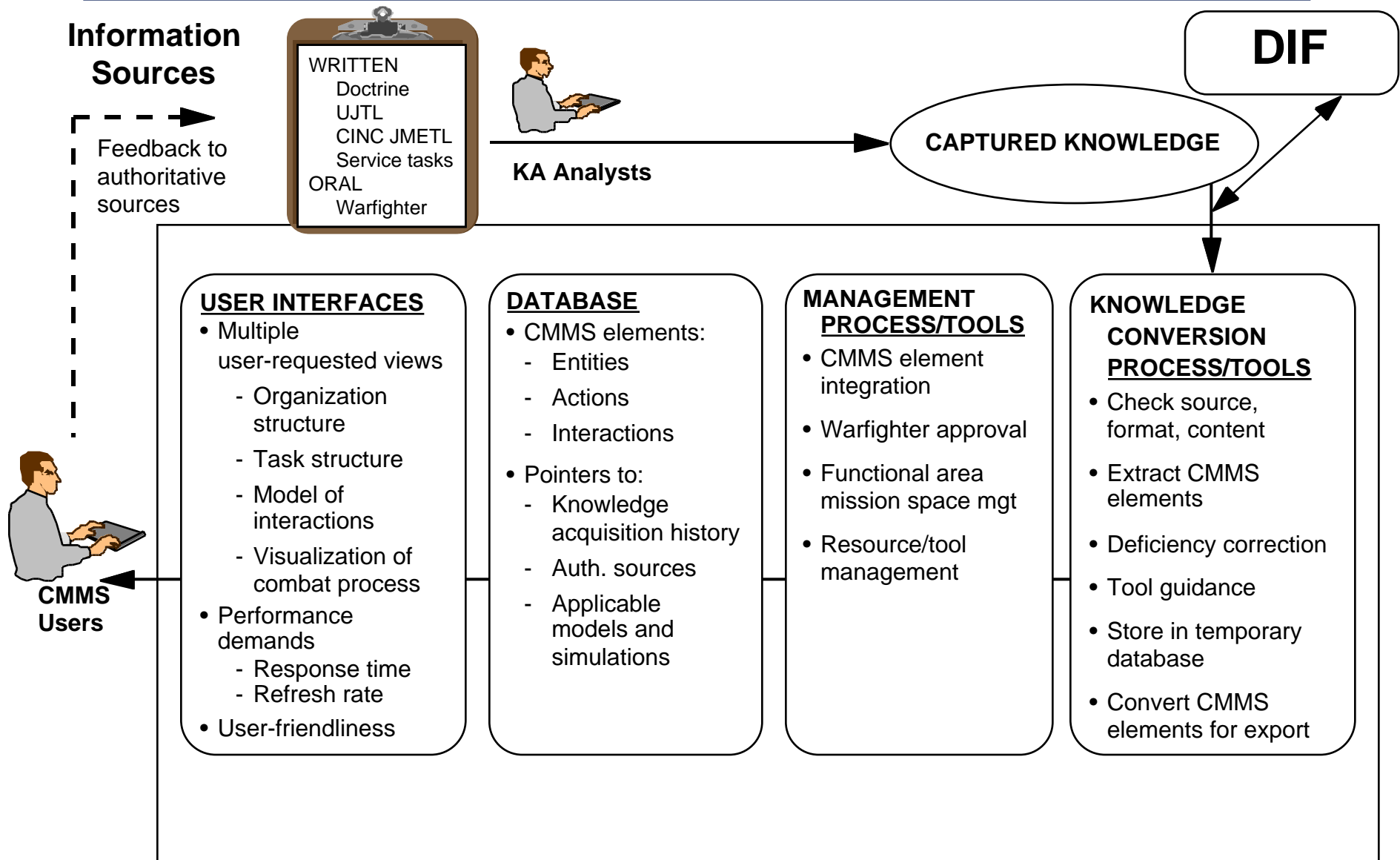
# Simulation Development Sequence



## CMMS Development Sequence

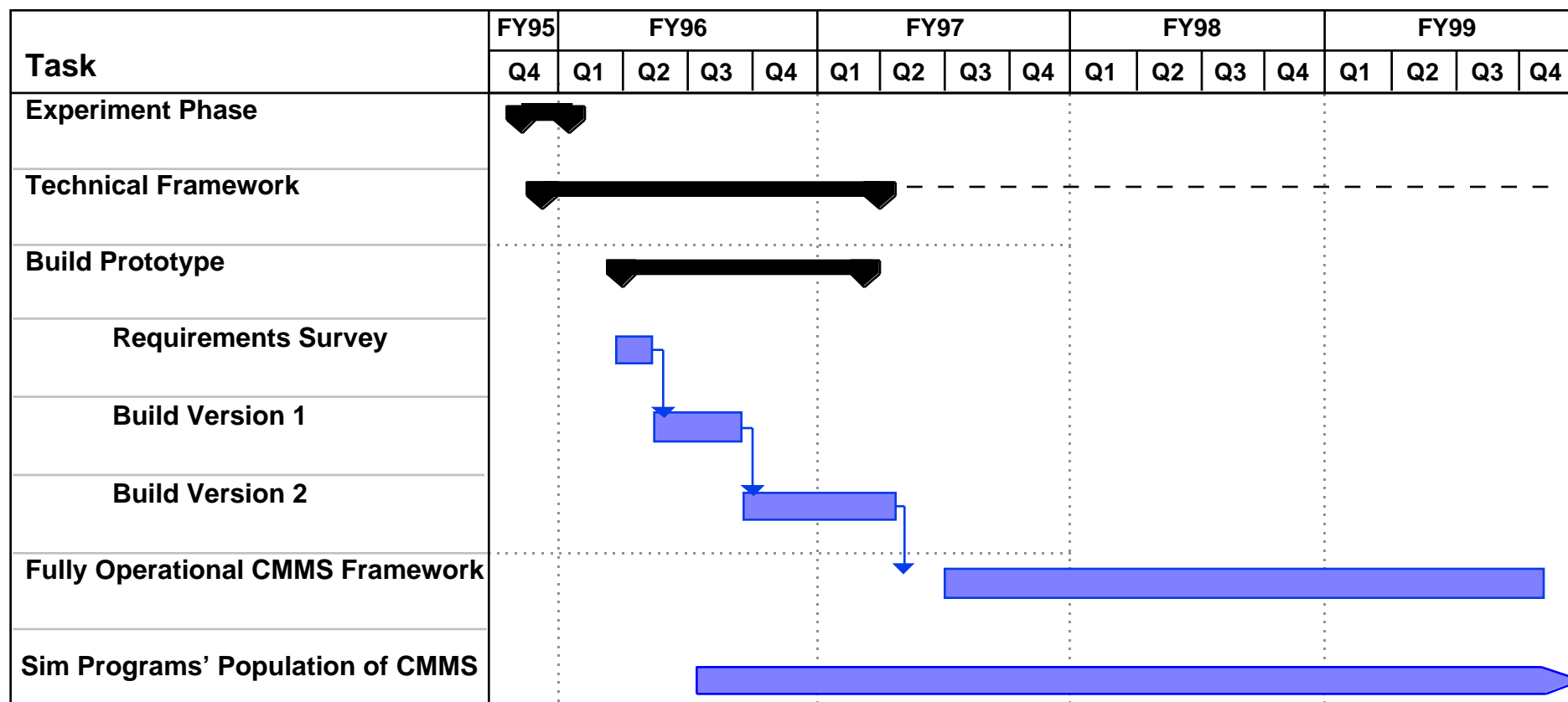


# CMMS Process/Tech Framework

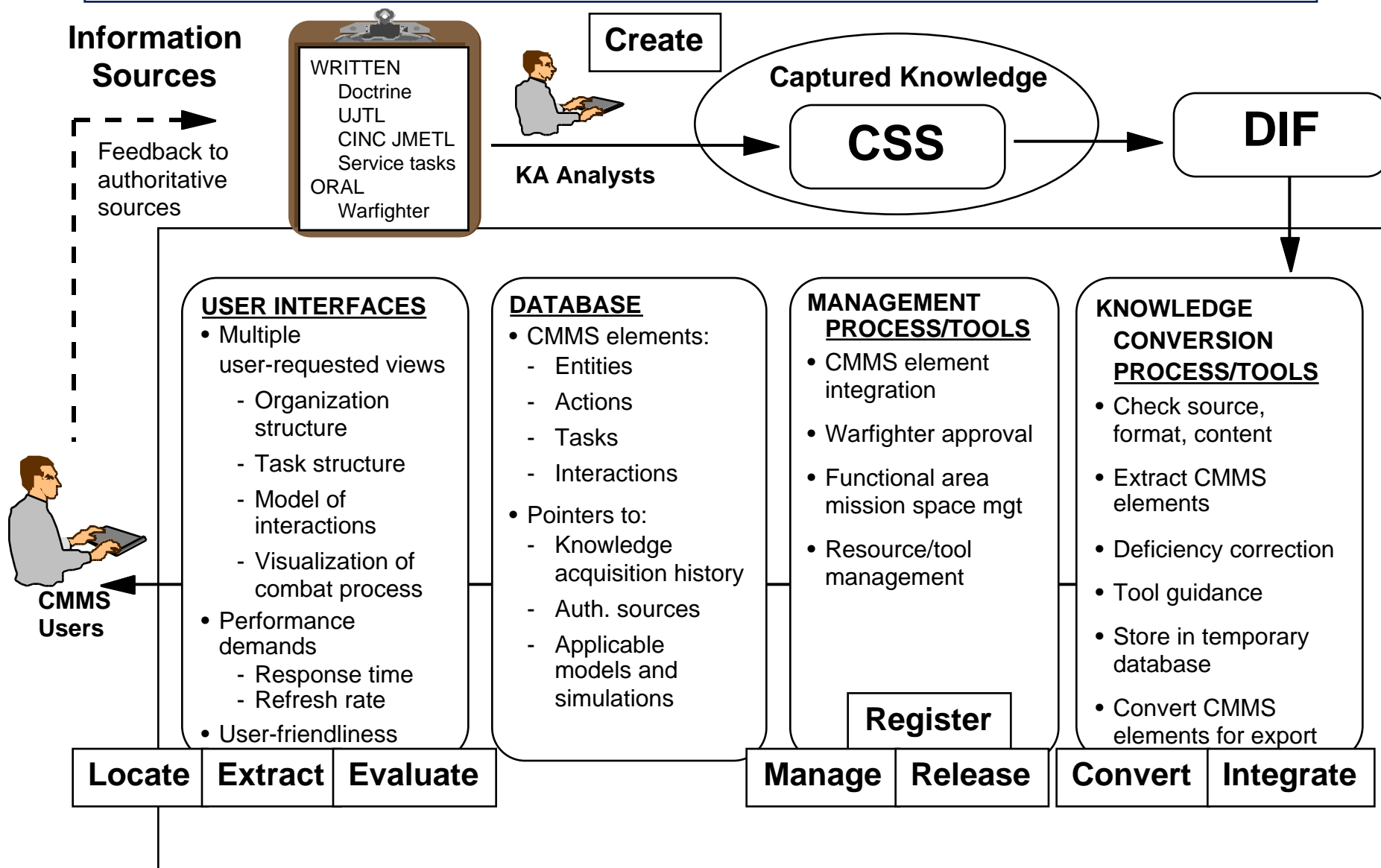


# Long Range Schedule

## CMMS Execution



# CMMS Prototype



## **Cycle 1 - Lessons Learned**

- **No CSS, no integration**
- **DIF required for efficient conversion**
- **Export API's necessary to compliment import API's**
- **Is VV&A done prior to conversion and integration, after, or both?**
- **How are VV&C values aggregated?**

## Cycle 2 - Enhancements

- **Develop DIFs and improved conversion routines for existing programs**
- **Add functionality to capture additional elements in Technical Framework (e.g., conditions, sequencing)**
- **Add functionality to reflect new Technical Framework functions**
- **CMMS use-case by simulation developer (UFR)**

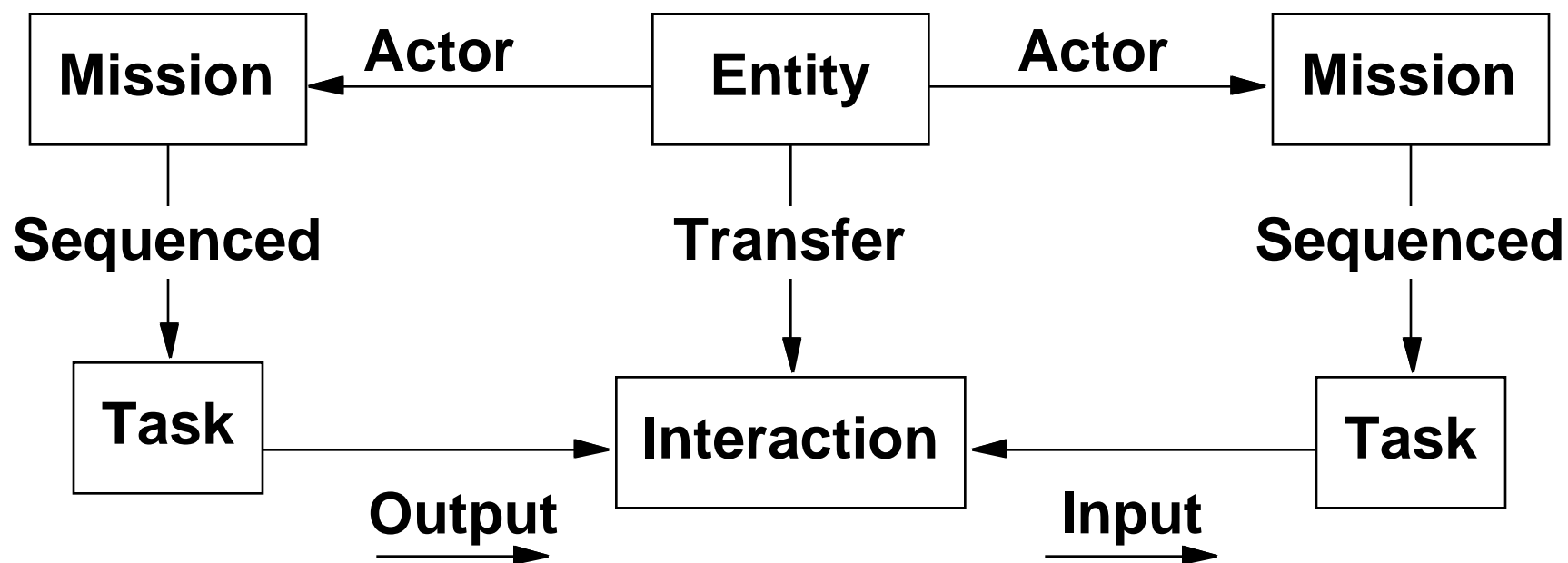


## **A Valid CMMS Is Essential**

- **Direct link from warfighter to simulation developer**
- **Makes simulation developers' job more efficient**
- **Results in simulations the warfighter can depend on**

# Backup

# Task-Interaction



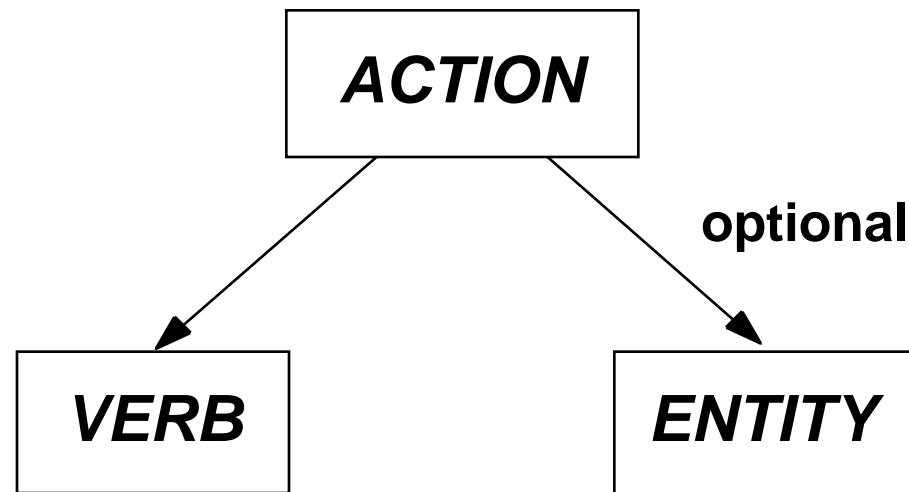
# ENTITY-Based Abstractions

- **ENTITY**     A distinguishable person, place, thing, or concept about which information is kept [2]. In particular, *ENTITY* includes the notions of person, organization, facility, feature, materiel, and plan defined in [5].
  
- **Concrete Examples**
  - **Person**            Wing Commander, Pilot
  - **Organization**   JFACC, Flight
  - **Facility**            Airbase, Power Plant
  - **Feature**            Road, River, Bridge
  - **Materiel**            F15, AWACS, Sidewinder
  - **Plan**                ATO
  
- **Abstraction Examples**
  - **Core C2**            GCCS entity-relationship diagram
  - **JWSOL**            JTF-ATD object classes

# ACTION-Based Abstractions

- **ACTION** The alteration or transformation by natural force or human agency which produces a change in state or condition. *ACTION = VERB [+ ENTITY]*
- **Concrete Examples**
  - **Physical Verbs** Move, Sense, Communicate, Engage, Replenish,...
  - **Cognitive Verbs** Develop, Monitor, Analyze, Supervise,...
  - **Capabilities**
    - Refuel Aircraft
    - Launch Missile
    - Detect Submarine
    - Generate ATO
- **Abstraction Examples**
  - **UJTL** Process-oriented operations templates
  - **CMMS Verb Syntax** Behavior-oriented C2 templates

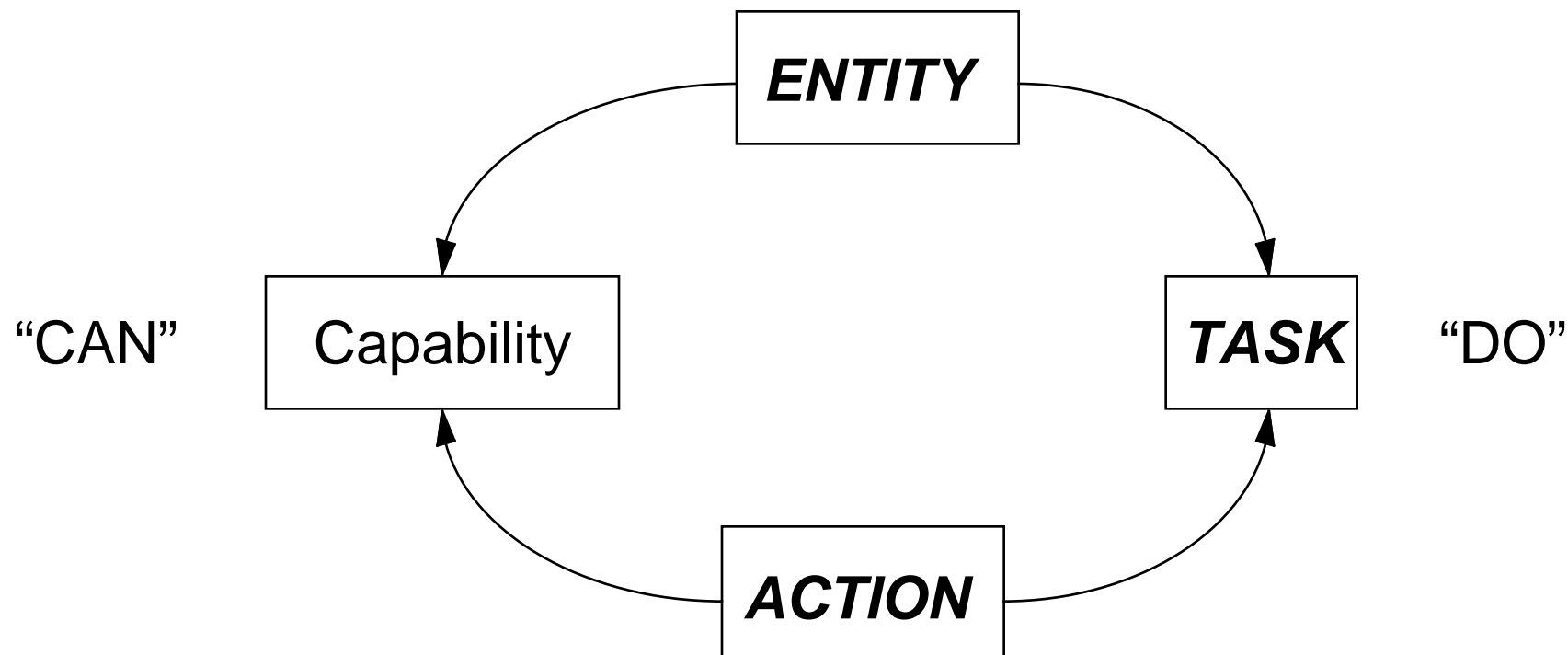
***ACTION = VERB + ENTITY* defining a Role**



# TASKS

- **TASK**                      The execution of one or more *ACTIONS* by an *ENTITY*.  
*TASK* is the smallest unit of unambiguous operational meaning.
- **Concrete Examples**
  - ENTITY                      ACTION (VERB + ENTITY)
  - KC-135                      Refuels F-15
  - F-15                          Launches AIM-9L
  - JFACC                        Generates ATO
- **Abstraction Examples**
  - Linguistics                      Subject-Verb-Object is a Sentence
  - Mathematics                      Domain-Mapping-Range is a Function

# ENTITY-ACTION-TASK-Capability





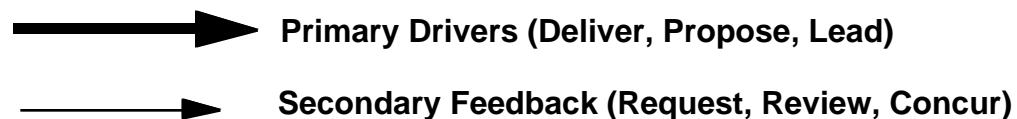
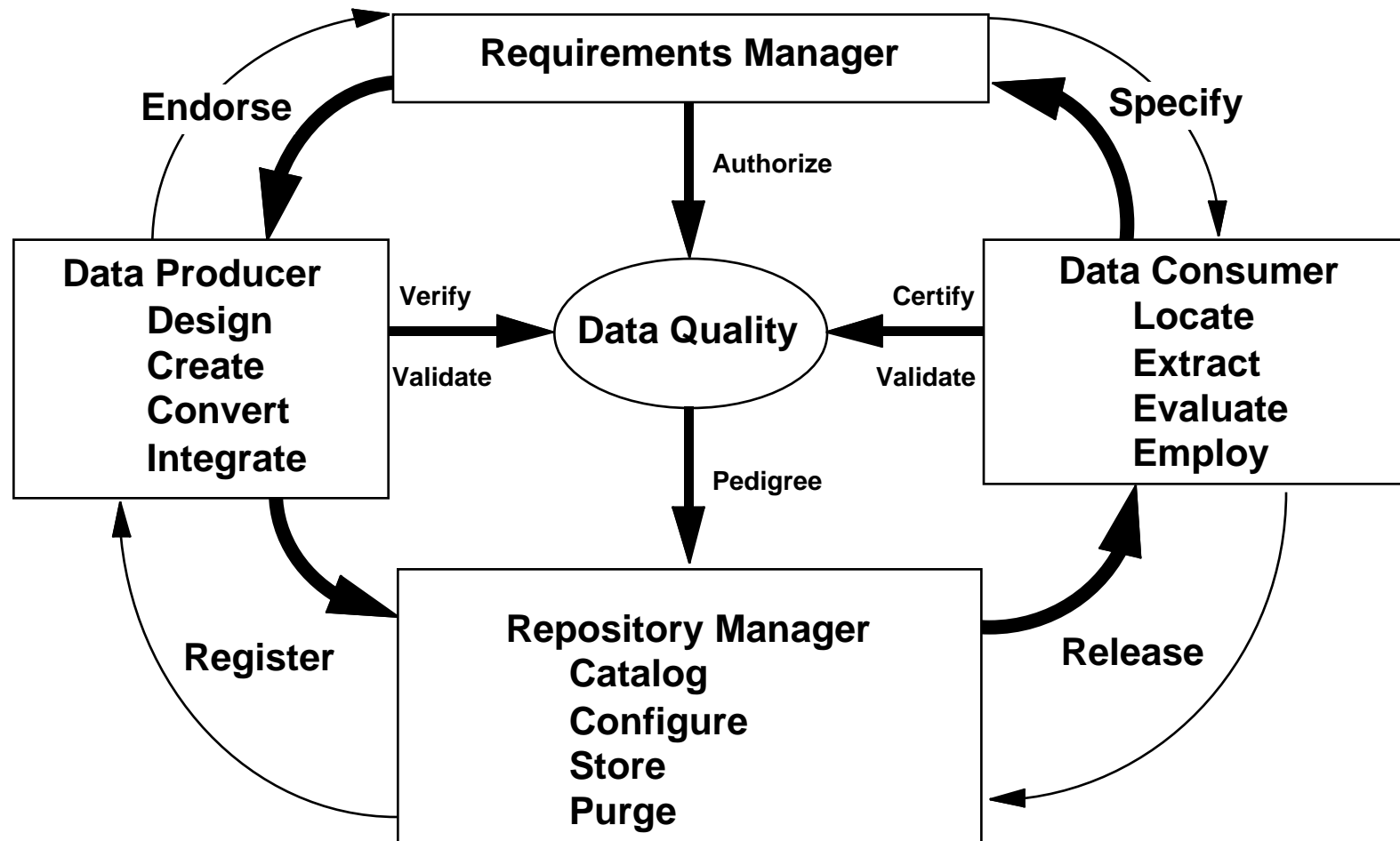
# INTERACTIONS

- **INTERACTION**      The interface which defines the flow of *ENTITIES* and/or *ACTIONS* between *ENTITIES* executing *TASKS*
  
- **Concrete Examples**
  - **TASK**                      Joint Targeting Board (Develops Joint Target List)
  - **INTERACTION**      JTB (Provides JTL) to JFACC
  - **TASK**                      JFACC (Develops ATO)
  
- **Abstraction Examples**
  - **State Transition Diagrams**      **ENTITIES** exchange state information
  - **Event Trace Diagrams**              **ACTIONS** initiated by events

# Mission-Mission Space

- **Mission**
  - A sequence of one or more tasks executed by an (actor) *ENTITY* to achieve a specific Objective or related set of Objectives
- **Mission Space**
  - A set of Missions which share a common organizing principle, purpose, or feature

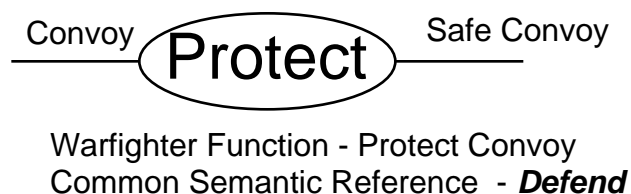
# Data Engineering Process, Version 0.1.3



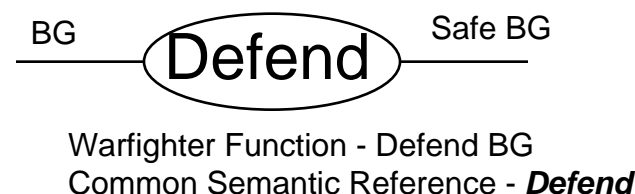
# Semantics Illustration

- Consider similar operations in two different contexts

## Army Context



## Navy Context

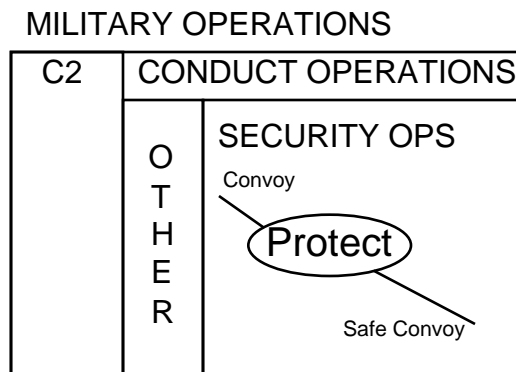


- Need to relate “Protect” to “Defend”
  - Establish a common underlying term for reference purposes
  - Establish a one-to-one correlation between terms and their accepted meanings or “senses”
  - Call it “Common Semantics”

# Syntax Illustration

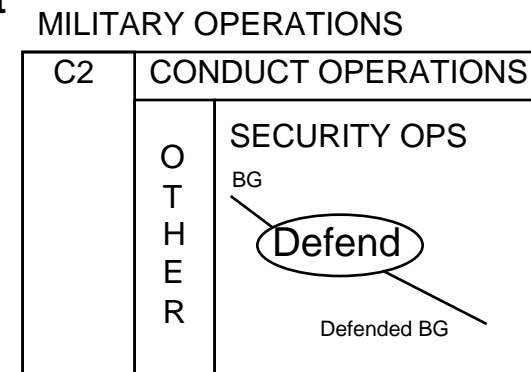
- Consider the word in context of the two operations

## Army Context



Warfighter Function - Protect Convoy  
 Warfighting Context - Security Operations  
 Common Semantic Reference - **Defend**

## Navy Context



Warfighter Function - Defend BG  
 Warfighting Context - Security Operations  
 Common Semantic Reference - **Defend**

- To be meaningful, common semantics must be allowed to differ in different contexts
- Need to establish the context for this sense of “Defend”
  - Establish a generic syntax or structure of contexts
  - The common semantic term is valid within one of these contexts
  - Narrows the field enough to find similarities
  - Call it “Common Syntax” (where the word may be used)

# CMMS Verb Syntax

## Command and Control

Monitor Situation

Mission/Task Analysis

Course of Action Development

Supervise and Synchronize

## Execution

Move

Sense

Communicate

Engage

Replenish

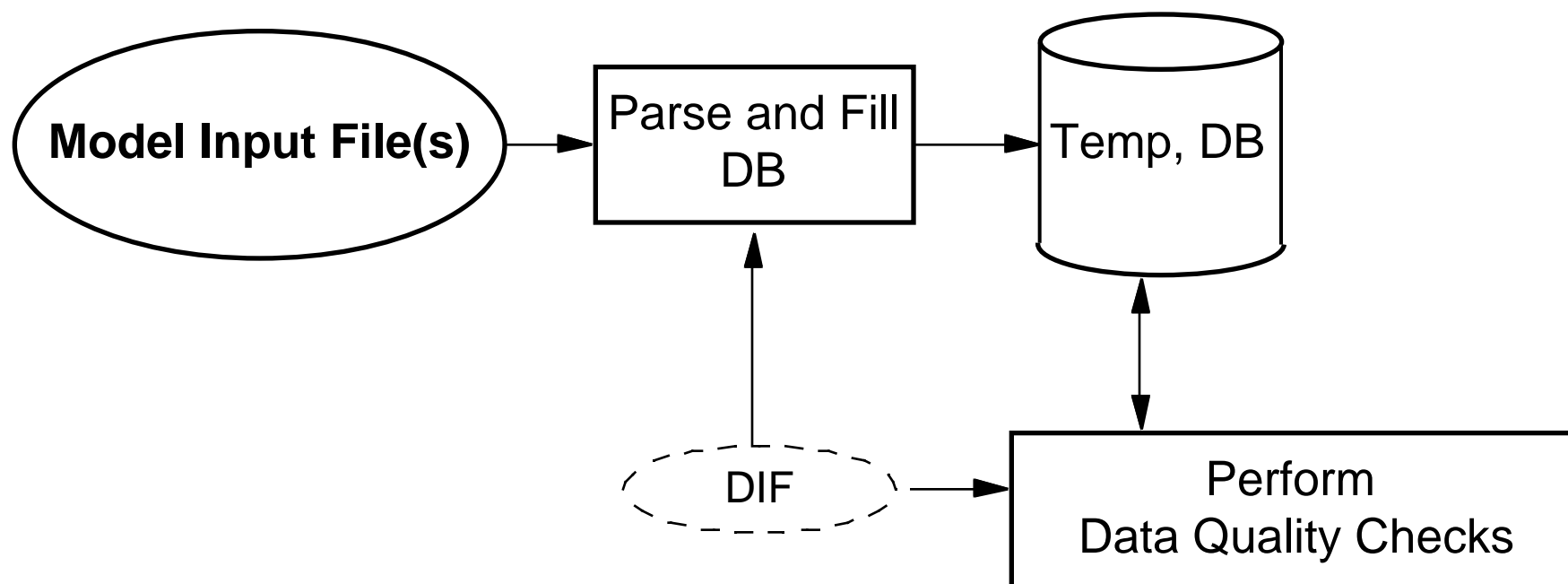
Repair

# Common Syntax and Semantics

- **Layer-0: CSS for General Purpose IS Templates**
  - Level-0 Unstructured Text, Freeform Diagrams
  - Level-1 IDEF0 Activity Model, IDEF1X E-R Diagram, Op-Spec
  - Level-2 Use-Case, Process, Behavior Diagrams (Core, Statemate, RDD)
  - Level-3 Booch, Rumbaugh, Schler-Mellor, UML, IDEF Object, ...
- **Layer-1: CSS for Structure and Content Specific to the Mission Space**
  - Level-0 Flat Lexicon (JP m-nn series)
  - Level-1 Global Namespace Data Dictionary (Core C2 data model)
  - Level-2 Hierarchical Semantics with Recursive Elements (CMMS EATI)
  - Level-3 Multiple-Inheritance Data Elements

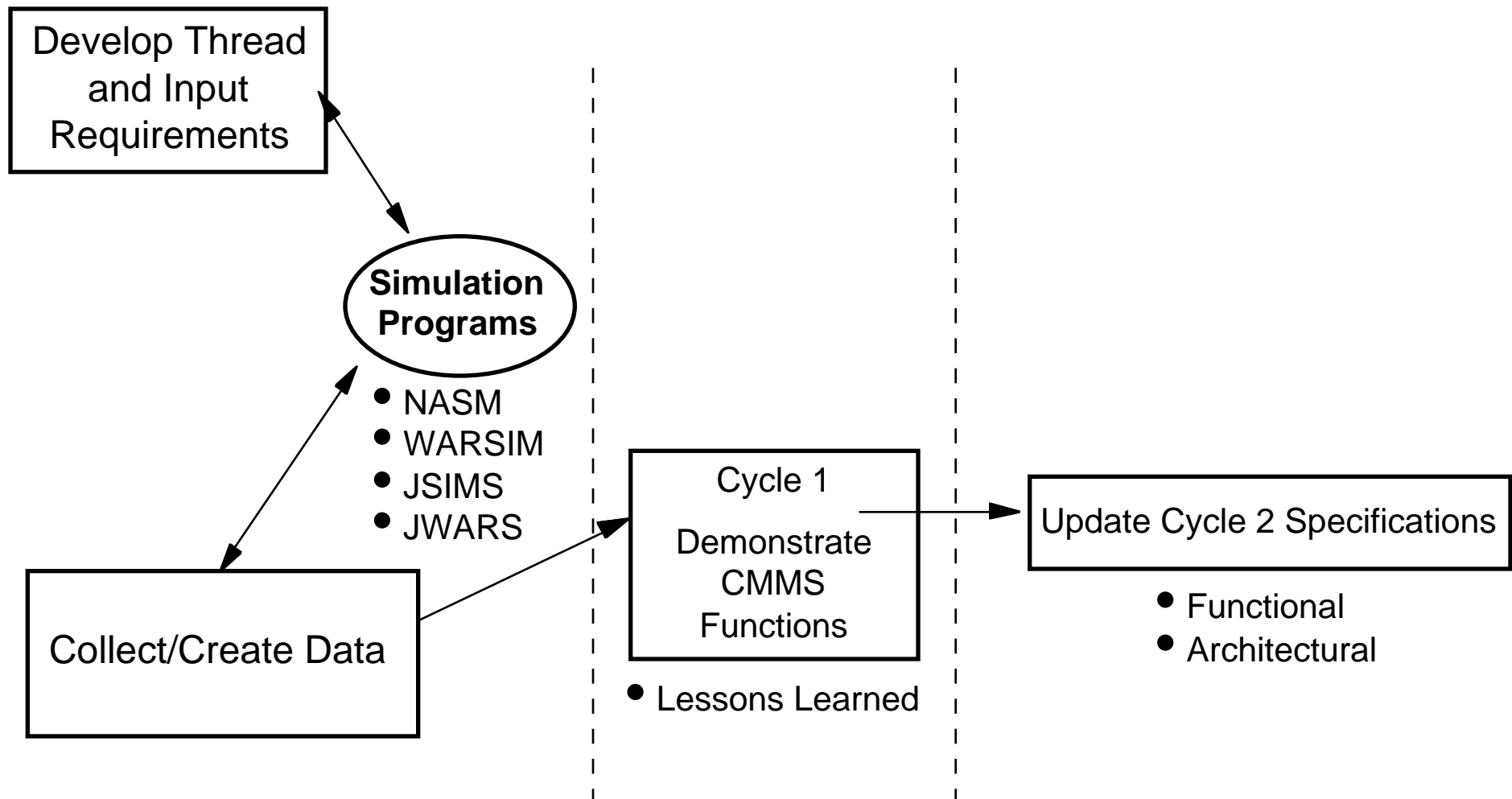
-----
- **Layer-2: CASE Tool-Specific Style Guides**
  - Level-0 Tool Features to Include/Exclude
  - Level-1 Tool Feature Usage Conventions
  - Level-2 Model Structure and Layout Conventions

# Cycle 1 Conversion Process





# Prototype Operational Concept



## **DRC Thread Data**

- **DRC Created 4 High Level Templates**
- **Each template based on UJTL task and JMETL template concept**
- **Lower level templates linked to simulation program data**
- **Templates**
  - **Conduct Joint Deployment (SecDef)**
  - **Protect and Support Forces (Unified Command)**
  - **Conduct Joint Interdiction (Sub-unified Command)**
  - **Conduct Air Interdiction (Air Component Commander)**
- **Thread implemented on Korean Interdiction scenario**

# ***Cycle 1***

## **Create**

**Create**    Capture of real world information about one or more Missions  
in the form of a Model for eventual inclusion in CMMS

- **Goal: Minimize DMSO sponsored knowledge acquisition**
- **Obtain/use knowledge from 4 existing programs**
  - JSIMS
  - WARSIM
  - NASM
  - JWARS
- **Obtain/create knowledge to support integrated mission thread**

# ***Cycle 1***

## **Data Dictionaries**

- **Two Dictionaries**
  - **Entities**
  - **Verbs**
- **Sources**
  - **JSIMS Data Dictionary**
  - **JWARS Data Dictionary**
  - **Excerpt from DMSO Verb Dictionary**
  - **NASM RDT Terms (derived)**
  - **WARSIM Terms (derived)**
  - **DRC Thread Data Terms (derived)**

# ***Cycle 1*** **Convert**

**Create**    Transformation of a Model which has been registered from its native form to a standard form required by CMMS including extraction of semantic and syntactic elements CMMS

- **Convert data for 4 programs**
  - **JSIMS**
  - **WARSIM**
  - **NASM**
  - **JWARS**

# ***Cycle 1***

## **Integrate**

**Create**    The act of combining, normalizing, storing, indexing, and in general migrating Registered Models in CMMS standard form to a higher level of structural maturity and semantic enforcement within a unified database CMMS

- **Perform relationship checks**
- **Link data to items in data dictionary**
- **Link new data to data already in database**

# ***Cycle 1***

## **Integration Process Special Checks**

- **Assign Nouns to Entities in Dictionary**
  - Look for matching name
  - Search for synonym with matching name
- **Assign Actions to Verbs in Dictionary**
  - Look for matching name
  - Search for synonym with matching name
- **Identify Links to Existing Entities**
  - If entity with same name already exists, do not add entity but link to that entity
- **Identify/Check Relationships**
  - Identify actions without actors
  - Identify inputs without sender action
  - Identify outputs without receiver action
  - Identify task interactions that do not interact properly with actions, entities, or roles

# ***Cycle 1*** **Register**

**Register**    Submission of one or more models for actual inclusion in CMMS including source, format, and content checking with deficiency correction as appropriate CMMS

## **Producer**

- Registers new model (e.g., JSIMS, WARSIM, NASM, JWARS) or updates an existing model
- Provides summary information on model
- Indicates privileges to be granted to users
- Describes location of model (model loaded)



# Cycle 1 Manage

**Manager**      The provision of configuration management, version control, change traceability, data storage, and resource allocation for Models in CMMS

- **Configuration Management**
  - All data tied to model and version
- **Traceability**
  - Interface provided for updating reference documents
  - VV&A pedigree updated to reflect reference changes
- **Data Storage**
  - Implemented via a COTS DBMS
  - Data structure consistent with EATI canonical form
- **Installation Instructions\***
  - Describe installation procedures and requirements including resources
- **Event Services**
  - Producers notified of Integration, consumers notified of model changes\*
  - Integration and release reports provided

\*Not Implemented in Cycle 1

# **Cycle 1 Release**

## **Release**

**The provision of security services, access control, user identification for use and examination of Models**

- **User Identification and Access Control**
  - **Users must be registered and authenticated**
  - **Users only allowed to perform functions and access models for which they have been authorized\***
- **Security Services**
  - **Cycle 1 implemented on C2 operating system**

**\*Not Implemented in Cycle 1**

# ***Cycle 1***

## **Locate**

### **Locate**

**The use of on-line browsing tools, automated searches, and retrieval queries to identify Models of interest**

- **Fully Structured Views (FSVs)**
  - **Provide Information Interaction FSV**
- **Provide Following On-Line Reports**
  - **Organization Mission Tasks**
  - **Task Data**
  - **Model Semantic Assignments**
  - **Model Data Quality Errors**
  - **Model Relationship Errors**

# ***Cycle 1*** **Extract**

## **Extract**

**The use of application programming interfaces and automated data retrieval services to gather, format, package, and delivery CMMS Models to simulation developers**

- **Provide on-line queries**
- **Provide capabilities for downloading data displayed in queries and reports**

# ***Cycle 1*** **Evaluate**

## **Evaluate**

**Determination of the suitability of a CMMS Model  
for specific end-use by a simulation developer**

**Provide user with on-line interface to update VV&A pedigrees  
(model version, entity, action, reference)**